IED STAFF REPORT

LARGE DATA SETS:

PROCESSING GUIDELINES FOR ECONOMISTS

September 2, 1980

Kevin J. Lanagan

No. 2-1981

Cop 2

International Economics Division · Economics and Statistics Service · USDA

This paper was prepared for limited distribution to the research community outside the U.S. Department of Agriculture. The views expressed herein are not necessarily those of ESS or USDA.

LARGE DATA SETS:

PROCESSING GUIDELINES FOR ECONOMISTS

September 2, 1980

Kevin J. Lanagan

No. 2-1981

Kevin J. Lanagan
International Economics Division
Economics, and Statistics Service
U.S. Department of Agriculture

500 12th Street S.W.
Room 390
Wash., D.C. 20250

September 2, 1980

LARGE DATA SETS: PROCESSING GUIDELINES FOR ECONOMISTS. By Kevin J. Lanagan:
International Economics Division; Economics and Statistics Service
U.S. Department of Agriculture. Washington, D.C., 20250.
September 2, 1980.

ABSTRACT

This paper lists guidelines for analysts working with large data sets intended for computer storage and manipulation. Particularly when processing very large data sets, organization and planning of data collection, data preparation, and loading are extremely important. Considerable loss of time and money may result from oversights such as inconsistency in naming conventions or variable scales. Even a handful of aberrant record formats in a data set of 2000 variables could later require extra time for set merging. Careful preparations helps avoid such errors and inconvenience.

Key words: computer, data processing, data storage, guidelines.

This paper was reproduced for limited distribution to the research community outside the U.S. Department of Agriculture. The views expressed herein are not necessarily those of ESS or USDA.

886532

LARGE DATA SETS: PROCESSING GUIDELINES FOR ECONOMISTS

Outline

Introduction

A. Assembling and Organizing Data
   1. General specification
      a. Variable names
      b. Series length and scale
   2. Data entry options
      a. Tape and disc
      b. Cards
      c. Interactive
   3. Entry preparation
      a. Tape and disc
      b. Cards
         (1) Coding sheets
         (2) Card numbers
         (3) Format corrections
         (4) Accuracy check
         (5) Completion by contract
      c. Interactive

B. Loading Data Onto System
   1. Data entry and correction
      a. Loading
         (1) Tape
         (2) Cards
         (3) Interactive
      b. Merging
         (1) Tape/Disc
         (2) Cards
         (3) Interactive
         (4) Review check
   2. Data transfer
      a. To permanent file
      b. To intermediate files
      c. Direct transfer to interactive use

Introduction

This paper provides a selected set of guidelines for ESCS analysts assembling and working with data sets of prodigious dimensions: 1000 records or more. Some guidelines apply to smaller data sets as well.

These guidelines are intended for analysts and economists, not computer scientists or programmers. A few of the procedures are tailored to users of the Speakeasy/Fedeasy interactive language.

Though recommendations may seem at times obvious, or methods annoyingly protracted, each one is presented with the conscious intention of helping analysts avoid common pitfalls of loading and manipulating very large data arrays. The cost of mistakes in time and expense is high when dealing with large data sets.

A publication is being planned by the Data Management Task Group--a joint effort of IED and DSC--which will summarize in more comprehensive fashion many guidelines for data processing, for large and small data sets. It is hoped that this paper provides a contribution to such pooling of experience and information on the use of computers for economic and statistical analysis in ESCS.

Note: It is hoped that the following suggestions and comments prove helpful to analysts. Nevertheless, appropriate programming and data management personnel should always be consulted prior to a major coding, processing, or storing exercize to insure that proper consideration is given to all options available to the analyst.

A.1.a. Variable names

A system or convention for naming variables in the data set should be selected prior to data entry. For convenienve in data handling and for greater ease in referring to individual variables, the variable naming convention selected should be uniform throughout the set.

IED currently recommends conforming to the OASIS (Outlook and Situation Information System) variable naming convention. Names for a wide variety of economic variables commonly used in agency analysis have been developed for OASIS. A brief illustrative table of examples follows this section. A more comprehensive list may be obtained from IED/World Analysis Branch. These variable names are specifically designed for use with foreign variables. Data series covering the same information for domestic analysis have been assigned different variable names. Note that all variable names contain seven characters.

If variables are commodity-specific (i.e., describe area harvested or volume imported of a particular agricultural item), then the seven-character OASIS variable name will contain three elements:

# of characters	element
2	1. commodity name
3	2. attribute (area harvested, imports)
2	3. country name

For example, RIAHHSN would represent rice area harvested in Senegal (RI = rice, AHH = area harvested, SN = Senegal).

If the variable names are not commedity-specific, the first five characters of the variable name are available to describe the attribute, and the variable name will therefore contain only two elements:

# of characters	element
5	1. attribute (population, GNP)
2	2. country name

For example, NIGNPSN would represent the gross national product for Senegal. (NIGNP = GNP for foreign variables, SN = Senegal). Variable names are not limited to literals, but can contain numbers. To distinguish the area harvested for two commodities beginning with the same letter (plantains, potatoes) the analyst might wish to specify the two variable names P1AHHSN (plantains) and P2AHHSN (potatoes). But in most cases an OASIS code will already be developed for commodity names.

Selected OASIS Variable Names
(Foreign Variables)

COMMODITY-SPECIFIC NAMES:

Area Harvested (wheat)	WHAHHXX
Production	WHSPCXX
Total Exports	WHUXTXX
Total Imports	WHSMTXX
Producer Price	WHFPCXX

NON-COMMODITY-SPECIFIC NAMES:

Total Land	NAITLXX
Arable Land	NIAARXX
Urban Population	DEPUBXX
Rural Population	DEPRLXX
Economically Active Population	WENEMXX
Literate Population	DELITXX
Primary/Secondary Enrollment	DEPPSXX
Research/Extension Workers	WENAEXX
Number of Tractors	FFQTRXX
Number of Draught Animals	FFQPAXX
Fertilizer Consumption	FEUDTXX
GNP	NIGNPXX
GDP:Agriculture	NIGDAXX
Private Consumption Expenditures	NICTLXX
Retail Price Index	CPOTLXX
GNP Deflator	NIDGNXX
Current Investment:Agriculture	FIETTXX
Capital Investment:Agriculture	FIECPXX

XX = Country code

A.1.b. Variable series length and scale

It is highly probable that the data set being assembled will include time series. Project analysts must agree upon a time period over which statistical analysis will be carried out, then data is collected taking this time period into account. Data series which cover only the time period studied may prove insufficient. For example, the period under study might be determined as 1965-78. If analysts later decide to run regressions using calculated three-year moving averages, they will lose the observations on either end of the series. A full series of such averages would require data over the period 1964-79. Rectifying ddta series of insufficient length at a later stage in a project proves unusually burdensome with large data sets, and is avoided by determining adequate data coverage at the outset.

Excluding special requirements, all data collected should be either already in or transformed to a uniform or compatible scale of measurement. If producer prices are expressed in local currency per metric ton, exchange rates should be likewise expressed in local currency per metric ton. Failure to catch incompatible scales typically results in meaningless or conflicting estimation results, and requires delaying analysis while executing scores of transformations which will prove annoying, time-consuming, and expensive.

A.2.a. Data entry options: Tape and disc

Data may be taken from storage files on computer tapes or discs. From the standpoint of error avoidance, this is much the preferred route. Unfortunately the analyst will rarely find that the data needed for a study is already conveniently collected and located on a tape or disc, ready for use. Nevertheless some analytical exercizes do draw from data sets--in ESCS or from outside sources--which are on tape or disc (e.g., ESCS Indices of Agricultural Production).

A.2.b. Data entry options: Cards

If all or a portion of the data set will be loaded from cards, the records must be punched individually onto the cards to be loaded. The same requirements of variable name and record length effective for data from tapes and discs (A.3.a.) apply also to data loaded from cards. Due to the need for accuracy, recommended preparations for card-punching are outlined in A.3.b.

A.2.c. Data entry options: Interactive

If all or a portion of the data set will be loaded interactively, records must be punched individually on the system and stored for later use. Variable name and record length requirements remain the same as for other data entry options.

The chief advantage of interactive entry over card entry (assuming no tape or disc already carries the data) is its one-step nature. The data is loaded immediately, completely sidestepping errors that could occur during the three stages of the card/batch loading method: coding, punching, and actual loading onto disc.

The major disadvantages would be (1) the relatively greater cost of interactive computer time, and (2) the inconvenience, impracticality, and potential inaccuracy of working at a terminal with all necessary data reference materials assembled together. This might require reading alternately among several large and detailed documents at one time.

A.3.a. Entry preparations: Tape and disc

Data to be loaded from tape or disc should be checked for compatibility with data being collected elsewhere in regard to two characteristics: variable names and variable record length.

Variable names are treated in A.1.a.

Variable record length (the number of distinct units of information in the variable) must be determined and fixed for a given data set, along with block size and other DCB specifications (computer jargon for descriptors of a data set). Within the constraints of that record length the analyst has flexibility in how individual observations are placed into the variable field and merged with other variables for analytical purposes. (If the record length is 10 but the variable has only 9 observations, the last space in the record field may be left blank.) But when a complete data set is merged with another set, the record lengths must be identical.

This compatibility can be easily achieved with the use of several software packages, among which are SAS, SPSS, and Speakeasy.

A.3.b. Entry preparations: Cards

Card-punching admits an enormous number of opportunities for error into the exercize of data base development. To avoid as many errors as possible, recommended preparations for card punching--or "coding"--are outlined below.

Information intended for computer cards must be encoded onto sheets designed for such a purpose. Most 80-column programming sheets, available from DSC, are suitable. Two problems typically occur when encoding a large number of variables.

1. Cell locations are not consistent. A variable record contains a variable name and other identifying information, and the data itself. The identifiers and data observations are placed into separate "cells" on the coding sheet. (A cell is any continuous string of column spaces forming a block for the observation. These cells are defined by specific beginning and ending column numbers.) Cells must be identical for all variables. If cell specification differs for even one record, the entire data set will have two distinct record formats, complicating seriously subsequent data merging.(The exception occurs when the coder enters data 'free form' in conjunction with one of the statistical packages such as SAS.)

2. Record continuity is lost when records extend over more than one card. Variables with ten or more observations normally require at least two 80-column cards to listall of the data. In such cases, observations for the variable will "spill over" to a second line on the coding sheet because one line cannot contain all of the observations. The second line has to be identified as part of the same variable begun in the first line, and cell definitions identically observed, to prevent inaccurate loading of the variable record. (The second line becomes a second "card" read by the system.)

To avoid these potential problems of data loading by cards, it is recommended that coding sheets be specially prepared before the coding process begins.

This preparation can be accomplished simply by placing heavy black lines between the columns that border individual cells, in order to clearly distinguish those cells. (Refer to Sample Coded Sheet #1 at the end of this section.) Note in the example that lines separate not only data cells, but also cells containing information which identifies the variable (variable name). Assuming the data is always right justified, clear distinction of cells insures that data observations will end at a specified column. In the sample, 1965 data ends at column 17 (1965 = 11) and 1972 data at column 66 (1972 = 555).

The reader will note in the example that the number of observations for the variable SJUXTBI "spilled over" into a second line. Guided by the same heavy line markings, the reader will find that 1975 data ends at column 17 (1975 = 100) and 1978 data at column 38 (1978 = 405). Once again, continuity of cell location is preserved.

Because cell length and record format remain the same for first and second cards, how are the two cards themselves distinguished? Within the variable field containing identifying information (columns 1-10) is a cell labelled "Card Number" (column 8). In this cell the card number 1 or 2--corresponding to first or second--is placed. Consequently observations for 1965 and 1975, though they always occur in the same cell bounded by columns 11-17, are distinguished because they correspond to different card numbers.

Note that the prepared coding sheet leaves columns 9-10 blank. This was done as a safeguard in case a need arises later to further specify each card with additional information.

After coding sheets are properly prepared, variables are entered. The coder should ensure that (1) all data are right justified within their cells; (2) zeroes or a comparable figure are entered where data is missing to prevent collapsing

of the record; and (3) each record uses the correct number of cards.

Coding Check

Once completed, all coding sheets should be checked to insure that cell locations are consistent for all of the variables in the data set. All coding sheets should then be reviewed--record by record--comparing coded data directly with sources to confirm their accuracy. It is recommended that a person other than the coder review the sheets. The probability of error with statistical coding--especially if several data sources have been used--is high. Someone not involved in actual coding might more readily catch inaccuracies.

The need for this data check, time-consuming though it will be, should be made clear. Many isolated errors can usually be spotted immediately in a file printout and easily corrected. Example:

INCORRECT: cornarea = 51 53 54 51 56 570 58 58 61 62 64 64

CORRECT: cornarea = 51 53 54 51 56 57 58 58 61 62 64 64

But if, for example, a coder either by oversight or ignorance used data for GNP at current market prices when coding the variable GNP at constant market prices, such an error would not be noticed by mere visual inspection when simultaneously reviewing 2000 variables, for countries with which the reviewer was not well acquainted.

INCORRECT: gnpcnst = 50 53 57 60 64 59 64 69 73 84 89 112

CORRECT: gnpcnst = 25 27 29 31 34 33 37 40 44 51 60 78

If the same coder entered all GNP data, the same error would most likely

have been made for all countries in the data set. Such an error, if unnoticed, would require numerous corrections at a later stage in the project (perhaps in the middle of complicated modelling exercizes) resulting in duplication of work as estimations are repeated with corrected data. *A check of coded data with sources, as recommended above, would catch errors of this sort at an early stage in the project*.

Card Punching

For large data sets, card punching of data already entered properly on coding sheets will most likely be contracted out to a private firm. The ESCS keypunching staff handles smaller data sets of 300-500 records for normal to fast turnaround time, or can take on larger jobs if less rapid turnaround time is acceptable.

Turnaround time on a contracted job of 2000-3000 records averages three days.

Care must be taken to make *perfectly explicit* on coding sheets such details as blank spaces, zeroes, and repeated entries. Keypunch contracting personnel should be left as little room as possible for interpretation. If repeated zeroes are required in a record, zeroes should be entered instead of cells left blank. If portions of variable names are repeated successively on a coding sheet, the entire variable name should be entered in full for every record, rather than substituting for them indications or marks such as *same*, *ditto*, ", or ↓. (See Sample Coding Sheet #2.)

U.S. DEPARTMENT OF AGRICULTURE
GENERAL PURPOSE CODING

PREPARED BY | PHONE NO | SYSTEM | PAGE OF

PROGRAM NAME | PROGRAM NO | DATE

COMMENTS

BLOCK FORM

SAMPLE 1

Commodity	Attribute	Country	Card Number	Blank	Data 1965 (1) 1975 (2)	Data 1966 (1) 1976 (2)	Data 1967 (1) 1977 (2)	Data 1968 (1) 1978 (2)	Data 1969 (1) 1979 (2)	Data 1970 (1)	Data 1971 (1)	Data 1972 (1)	Data 1973 (1)	Data 1974 (1)
SJ	UXT	BI	1		11	17	324	770	0	32	646	555	777	445
SJ	UXT	BI	2		100	321	675	405	0					

Variable name "SJUXTBI": SJ=Sugar Cane; UXT=Exports; BI=Burundi

SAMPLE 1 Continued

INSTRUCTIONS FOR CODING DATA

CARD NUMBER 1

			EXAMPLE
Commodity Code	Card Columns	1 - 2	SJ
Identifier	Card Columns	3 - 5	UXT
Country Code	Card Columns	6 - 7	BI
Card Number	Card Column	8	1
Blank	Card Columns	9 - 10	
1965 Data	Card Columns	11 - 17	11
1966 Data	Card Columns	18 - 24	17
1967 Data	Card Columns	25 - 31	324
1968 Data	Card Columns	32 - 38	770
1969 Data	Card Columns	39 - 45	0
1970 Data	Card Columns	46 - 52	32
1971 Data	Card Columns	53 - 59	646
1972 Data	Card Columns	60 - 66	555
1973 Data	Card Columns	67 - 73	797
1974 Data	Card Columns	74 - 80	445

NOTE:

Right Justify All Data
If Data Is Missing Put A Zero
Each Record Must Have 2 Cards

CARD NUMBER 2

Commodity Code	Card Columns	1 - 2	SJ
Identifier	Card Columns	3 - 5	UXT
Country Code	Card Columns	6 - 7	BI
Card Number	Card Column	8	2
Blank	Card Columns	9 - 10	
1975 Data	Card Columns	11 - 17	100
1976 Data	Card Columns	18 - 24	321
1977 Data	Card Columns	25 - 31	675
1978 Data	Card Columns	32 - 38	405
1979 Data	Card Columns	39 - 45	0

A B C D E F G H I J K L M N O P Q R S T U V W X Y

U.S. DEPARTMENT OF AGRICULTURE
GENERAL PURPOSE CODING

PREPARED BY

PROGRAM NAME

COMMENTS

SAMPLE 2

THIS:

1	P24XTCM1	10	10	30	10
2	P24XTCM2	57	31	45	45
3					
4	P24XTCI1	630	410	660	990
5	P24XTCI2	684	342	266	100
6					
7	P24XTKE1	7590	8700	5640	2470
8	P24XTKE2	982	3762	3114	3100
9					
10					
11					
12					
13					
14					

A B C D E F G H I J K L M N O P Q R S T U V W X Y

U.S. DEPARTMENT OF AGRICULTURE
GENERAL PURPOSE CODING

PREPARED BY

PROGRAM NAME

COMMENTS

NOT THIS:

1	P24XTCM1	10	10	30	10
2	CM2	57	31	45	45
3					
4	CI1	630	410	660	990
5	CI2	684	342	266	100
6					
7	KE1	7590	8700	5640	2470
8	KE2	982	3762	3114	3100
9					
10					
11					
12					
13					
14					

Variable Name "P2UXTCM": P2=White Potatoes; UXT=Exports; CM=Cameroon.
Variable Name "P2UXTCI": P2=White Potatoes; UXT=Exports; CI=Ivory Coast.
Variable Name "P2UXTKE": P2=White Potatoes; UXT=Exports; KE=Kenya.

A.3.c. Entry preparations: Interactive

Section A.2.c. listed the disadvantages of cost and potential inconvenience related to interactive data entry. Nevertheless, costs can be cut and data entry facilitated by following a few simple techniques in preparation for terminal entry.

1. Use photostat copies of reference material whenever possible and sensible, to avoid the necessity of positioning and securing large and unwieldy texts. This is particularly necessary when data originates in many disparate sources rather than one source. Flat pages are easier to sort and read.

2. Use visual guides (lines, colored pencils) to highlight the series or numbers desired.

3. Group all data sheets by country, commodity, attribute, or other category to insure organization.

4. Check all variables to be entered to insure that required years are present, eliminating needless searching and backfilling while entering data.

5. Construct and maintain a checklist of all variables intended for entering. A simple matrix table form is usually suitable. This will prevent uncertainty during data entry about what variables have been entered, and eliminate double-entry.

Note: These points constitute preparation that should be completed before accessing the computer system--preparation that can save appreciable TSO connect cost expense.

B.1.a. Data entry: Loading

Partitioned data sets empower analysts to request individual variables from the system. Sequential data files do not. Ordinarily when data is loaded into a computer system, it is done directly onto a disc where it is stored in a sequential file, which is relatively cheap because it uses less storage space. Then, if desired, data can be transferred to a partitioned file. This type of storage takes up more storage space because each variable must occupy a distinct record space in the file so the system can "recognize" or locate and retrieve it.

For data loading, the entire data set should be loaded and made available on disc in a sequential data file, to allow computer programmers to begin transfer of data to partitioned data files.

If the data is already on tape(which generally carries proper technical descriptive information about the format of the data) then the tape merely needs to be backed off onto a disc file. DSC can assist in this exercize.

If the data is already carried on disc, it is likely that nothing need be done until the data is to be transferred to a partitioned file. Insure that the record format of data on the disc matches the record format for other data, whether from cards or entered interactively.

If the data is to be loaded from cards (which we now assume are punched), the cards can be loaded directly onto a disc file (sequential).

(Example: Refer to Sample Coding Sheet #1. Card number 1 would be merged with card number 2, so that data from 1965-74 and data from 1975-78 would be joined to form a continuous variable record from 1965-78; the time series desired for analysis.) This merging is accomplished with a computer program, of which DSC can assist in preparation.

If the data is to be "loaded" interactively, every variable may be saved directly into a partitioned data set, short-circuiting the need for an interim sequential file. No programs are needed to transfer such data between files unless for special purposes, and the variables would be available immediately for analysis. Keep in mind that loading interactively often incurs significant expense (particularly when very large data sets are involved) a factor which must be weighed against the advantages of time-saving and convenience.

As data is entered interactively, save it into temporary or permanent storage periodically (every 10-20 variables), to guard against loss due to system malfunction. But *before* saving, tabulate each variable once after entering for visual check to see that the entire series is entered. This check is facilitated by tabulating variables in small groups to make improperly truncated variables more apparent. as the following page illustrates.

A visual check of the following variables shows that variable "WHSPCSD" is missing the last observation. This omission would not be as clearly evident in an ordinary Speakeasy display of a variable without a tabulation request.

```
WHSPCET  WHSPCKE  WHSPCSD  WHSPCTZ
•••••••  •••••••  •••••••  •••••••
  722      172       56       23
  739      128       69       39
  760      162       78       31
  782      216       88       44
  808      242      123       39
  840      222      115       61
  876      206      163       84
  782      164      140       98
  545      150      152       78
  500      149      236       46
  734      158      269       56
  507      176      264       58
  592      180      298       35
  423      144      317       28
  390      125                32
                    ▲
```

Several procedures exist within Speakeasy for correction of this omission:

```
WHSPCET  WHSPCKE  WHSPCSD  WHSPCTZ
•••••••  •••••••  •••••••  •••••••
  722      172       56       23
  739      128       69       39
  760      162       78       31
  782      216       88       44
  808      242      123       39
  840      222      115       61
  876      206      163       84
  782      164      140       98
  545      150      152       78
  500      149      236       46
  734      158      269       56
  507      176      264       58
  592      180      298       35
  423      144      317       28
  390      125      150       32
                    ▲
```

B.1.b. Data entry: Merging

At this stage of data entry the analyst has data "on the system" in one or more disc files, developed with data loaded interactively, or from cards, tapes, or disc. If all data is loaded from one source, no merging is necessary. If data is loaded from two or more sources, it is likely that the data will be loaded and saved in separate disc files. Separate files must be merged to form one cohesive data file with a uniform record format.

This merging step requires a program, which would be developed by DSC to suit the requirements of the specific files being merged. Writing, testing, and running of such a program requires about one week. (If files are already online and in the same format, this step is not required.)

The merged file shoudl then be run on a printout for quality check. This check will be the analyst's first opportunity to view the data as a complete set and spot errors. The entire data set should be scrutinized. Common errors to be alert for at this stage include:

(1) duplicated variable names with identical data;

(2) duplicated variable names with different data;

(3) variables not belonging in the set;

(4) variables missing data partially or entirely;

(5) coding errors obvious by visual inspection
("cornarea" example in A.3.b.)

General comments on data transfer

The data set is now presumed to be complete and located in a sequential data file. The analyst will normally want the data available for interactive use. Transferring the data from this sequential file to a partitioned data file allows such use. Analysts in IED will most likely want to transfer data to partitioned data files amenable to the Speakeasy/Fedeasy interactive language, which are called *mykeep* files.

Such data transfer presents problems related to computer processing time, space allocation, and expense. The analyst should therefore consider data transfer options discussed hereafter. These options are summarized visually in flow chart form at the end of this section.

B.2.a. Transfer to permanent file

Transfer the entire data set simultaneously into one mykeep file.

If the data set contains over 1500 variables, this option would be best accomplished by batch.(Interactive transfer could bog down if the system is in peak use, resulting in large connect charges.)However, even batch transfer can develop expensive complications unless certain errors are expressly avoided:

1. Within the job commands submitted to execute the transfer (in preparation of which DSC will normally assist), specify NONUM for the mykeep file being filled. The NONUM command orders that the file being filled not be numbered. Otherwise the system will default by numbering the file to be filled. A job which attempts to copy a non-numbered data file (sequential sets are normally non-numbered) into a numbered file will not process, and may enter a loop with resultant system delays., or may destroy the data entirely.

2. Specify the proper time, space, and class to insure that the job will execute completely. For example, a data file of 2000 variables should be allowed approxemately 10 minutes and 100 tracks in Class F. If too little time or space is specified, the job will terminate before transfer of all variables is completed.

B.2.b. Transfer to intermediate files

Transfer the data set one section at a time.

This method is used if the entire set must be transferred quickly. It can be accomplished interactively without incurring undue expense by transferring small groups of variables (300-500) into separate *mykeep* files. Small groups are transferred in an interactive mode (taking care to allow proper space and time) *one/group at a time* into separate mykeep files, so the analyst or programmer can closely monitor the expense. If because of heavy system use, even transfer of small sets incurs high cost, the analyst can immediately switch to batch transfer, requiring more time but at a lower cost.

When the small data groups are transferred, the separate mykeep files into which they were transferred should be merged into one mykeep file. This file is then available for interactive use.

B.2.c. Direct transfer for interactive use

Transfer very small sets directly into working storage.

This method is used only if the analyst must use portions of the data before the entire set can be transferred into a partitioned file.

A program must be prepared which obtains variables from the disc file and brings them into working storage.

A note of caution regarding use of this procedure is in order. It should be employed only as an interim measure to allow short-term analysis while the full data set is being transferred. Reasons: (1) this method reads variables/into working storage only one at a time, a time-consuming manner of obtaining from disc more than only a very few variables; (2) the interactive structure of variables obtained with such a program could prove cumbersome, as the following example illustrates.

Example: The analyst wishes to consider data for production and area harvested of corn. One variable--corn area harvested--is on disc, while the other is not. The analyst uses the program described above to obtain corn area harvested ("cornarea") from disc, while corn production data must come from an outside source.

The program obtains "cornarea." Using Speakeasy, the analyst requests the area variable in working storage,

```
:_cornarea
CORNAREA (A 1 X 15 dimension real array)
51 53 54 51 56 57 58 61 62 64 64 65 67 67
```

then keys in data for production,

```
:_cornprod=104 110 110 108 114 116 116 117 123 123 126 126 127 130 131
```

and requests the production variable in working storage.

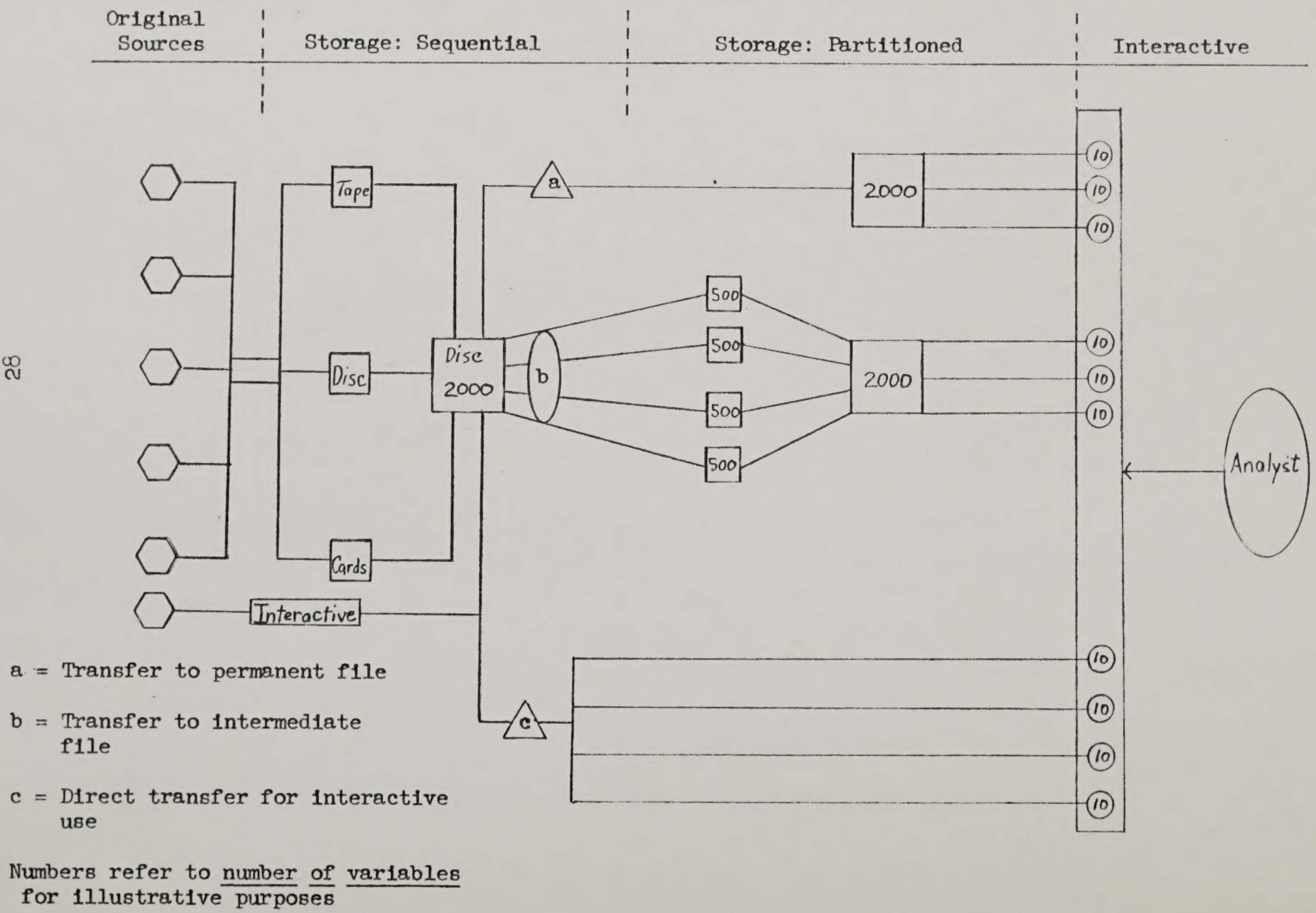
```
:_cornprod
CORNPROD (A 15 element array)
104 110 110 108 114 116 116 117 123 123 126 126 127 130 131
```

These two variables are incompatible in their present form for all computational purposes in Speakeasy. The reason is that though both variables are one-dimensional arrays of the same length, Speakeasy identifies them differently. The reader will note that one is a simple element array, the other an extract from an n X 15 array. This difference is due to the procedure by which the direct transfer program reads data into Speakeasy.

To make these variables compatible for computational purposes, the analyst would have to perform a transformation on one of the variables, as follows:

```
:_cornarea=trans(cornarea)
:_cornarea
CORNAREA (A 15 element array)
51 53 54 51 56 57 58 61 62 64 64 65 67 67
```

FLOW CHART FOR DATA TRANSFER OPTIONS


Original Sources
Storage: Sequential
Storage: Partitioned
Interactive
Tape
Disc
Cards
Interactive
Disc 2000
a
b
c
500
500
500
500
2000
2000
10
10
10
10
10
10
10
10
10
10
Analyst
a = Transfer to permanent file
b = Transfer to intermediate file
c = Direct transfer for interactive use
Numbers refer to number of variables for illustrative purposes